

Modely systémov

Abstraktné modely systémov,
ktorých požiadavky modelujeme

Podľa:

Sommerville – Software Engineering, 6th Edition, 2000

- objasniť, prečo by sme mali v rámci procesu RE modelovať aj kontext systému
- opísať behaviorálne, dátové a objektové modelovanie
- zaviesť notáciu používanú v jazyku UML
- ukázať použitie CASE nástrojov pri modelovaní systémov

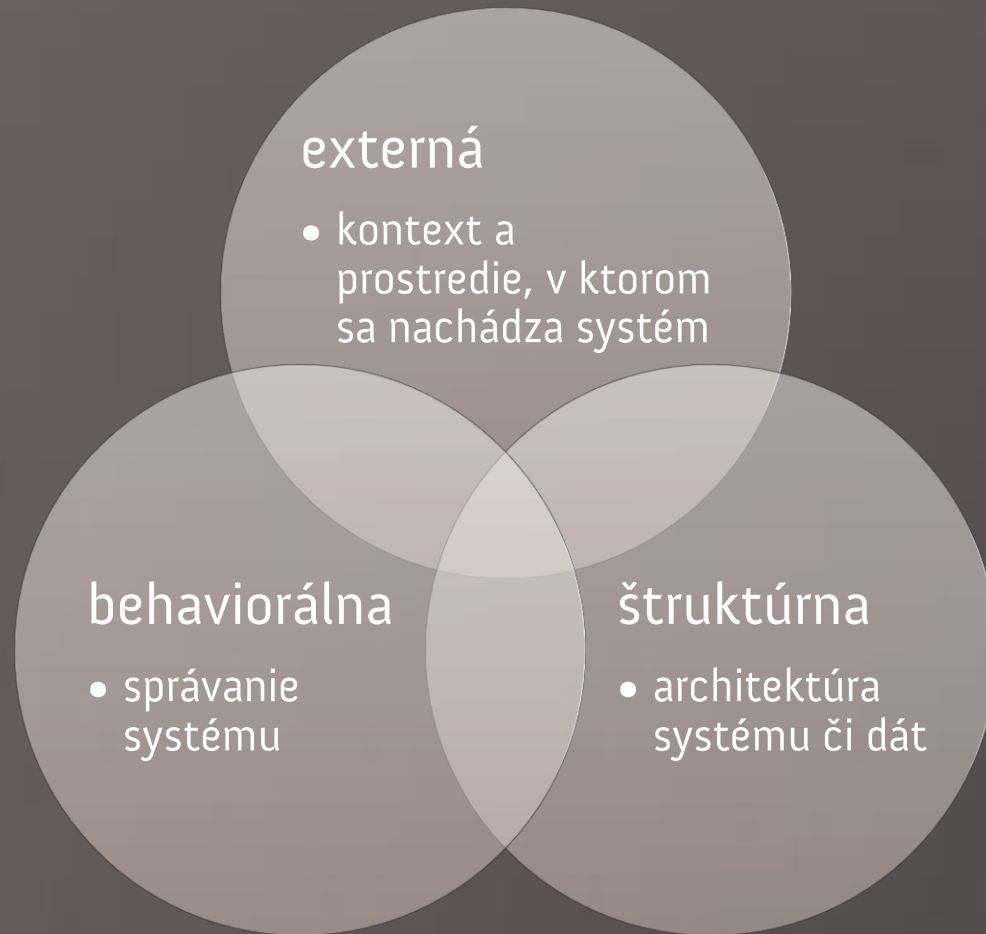
Pokryté oblasti

- modely kontextu
- behaviorálne modely (modely správania)
- dátové modely
- objektové modely
- modelovacie nástroje CASE

Systemové modelovanie

- pomáha analytikovi pochopiť **funkcionalitu** systému
- modely sú využívané v rámci **komunikácie** so zákazníkom
- rozličné modely umožňujú vidieť systém z rozličných perspektív

Rozličné perspektívy pohľadu na systém



Štruktúrované metódy

- zahŕňajú v sebe systémové modelovanie
 - je to ich prirodzenou súčasťou
- definujú
 - sadu modelov
 - proces, ktorými možno modely získať
 - odporúčania, ktoré sa vzťahujú na modely
- v CASE nástrojoch je systémové modelovanie podporované v rámci štruktúrovanej metódy

Slabiny štruktúrovaných metód

- nemodelujú nonfunkcionálne systémové požiadavky
- zvyčajne neobsahujú informáciu o tom, či je metóda vhodná pre riešenie konkrétneho problému
- môžu vygenerovať priveľa dokumentácie
- niekedy sú až nadmieru podrobné
- používatelia môžu mať problém s ich pochopením

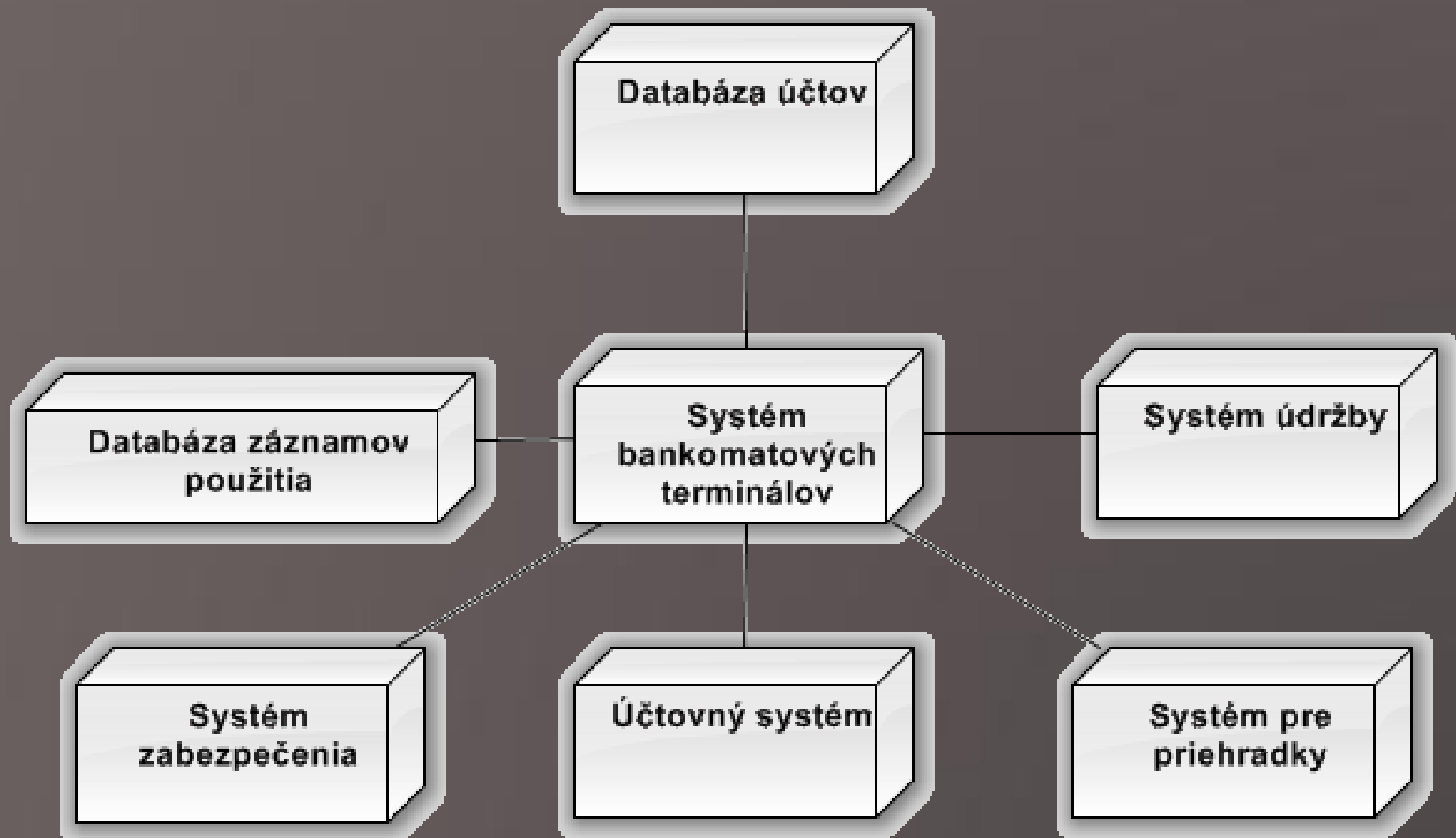
Analýza môže vyprodukovať modely:

Model	Anglický termín	Význam
Model spracovania dát	Data processing	Ako sa spracovávajú dáta v jednotlivých fázach?
Kompozičný model	Composition	Ako sú vytvárané entity z iných entít?
Model architektúry	Architectural	Najdôležitejšie subsystemy
Klasifikačný model	Classification	Spoločné črty entít?
Model stimulov/reakcií	Stimulus/Response	Reakcie systému na udalosti/podnety

Modely kontextu

- ilustrujú hranice systému
- tie môžu závisieť na organizačných či sociálnych hľadiskách
- modely architektúry ukazujú systém vo vzťahu k ostatným systémom

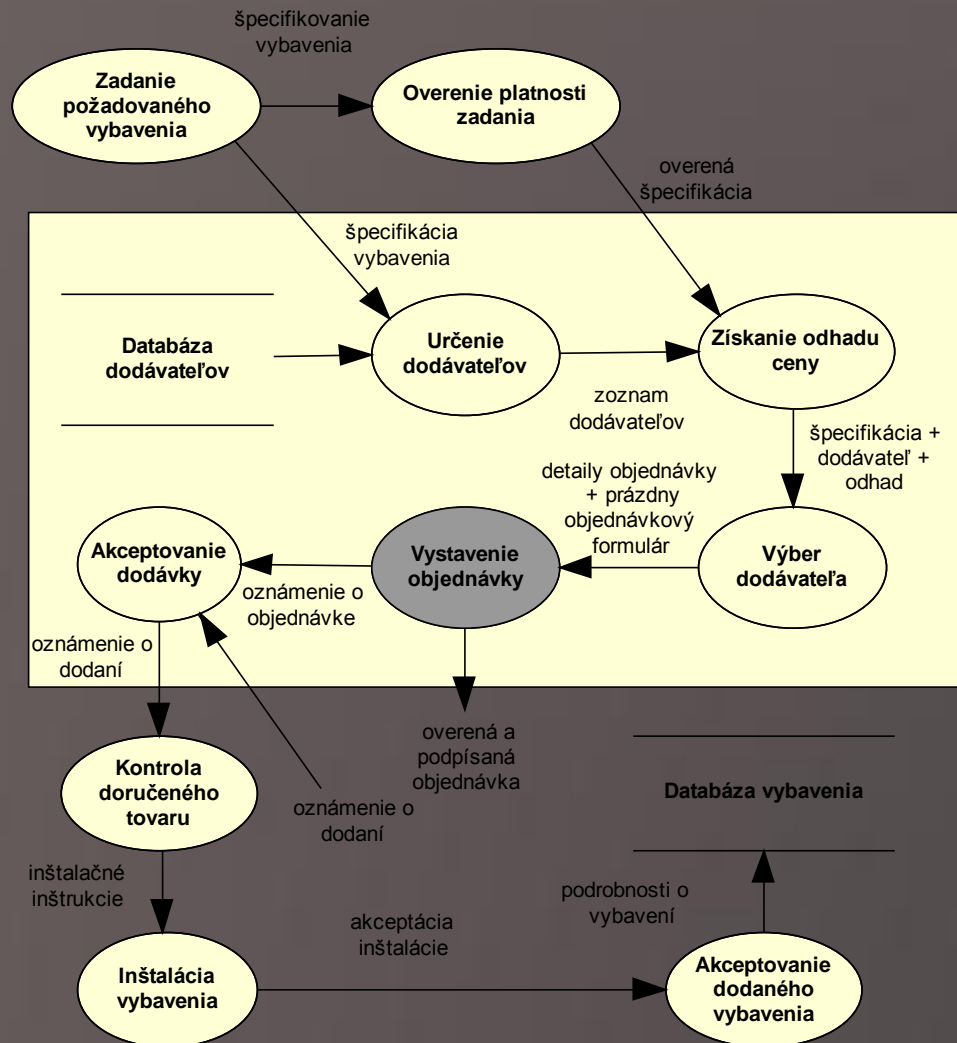
Kontext bankomatového systému



Procesné modely

- ukazujú celkový proces i čiastkové procesy, ktoré sú v systéme podporované
- modely tokov dát (**data flow** m.) možno použiť na zobrazenie procesov a toku informácií medzi procesmi

Príklad: proces získania vybavenia



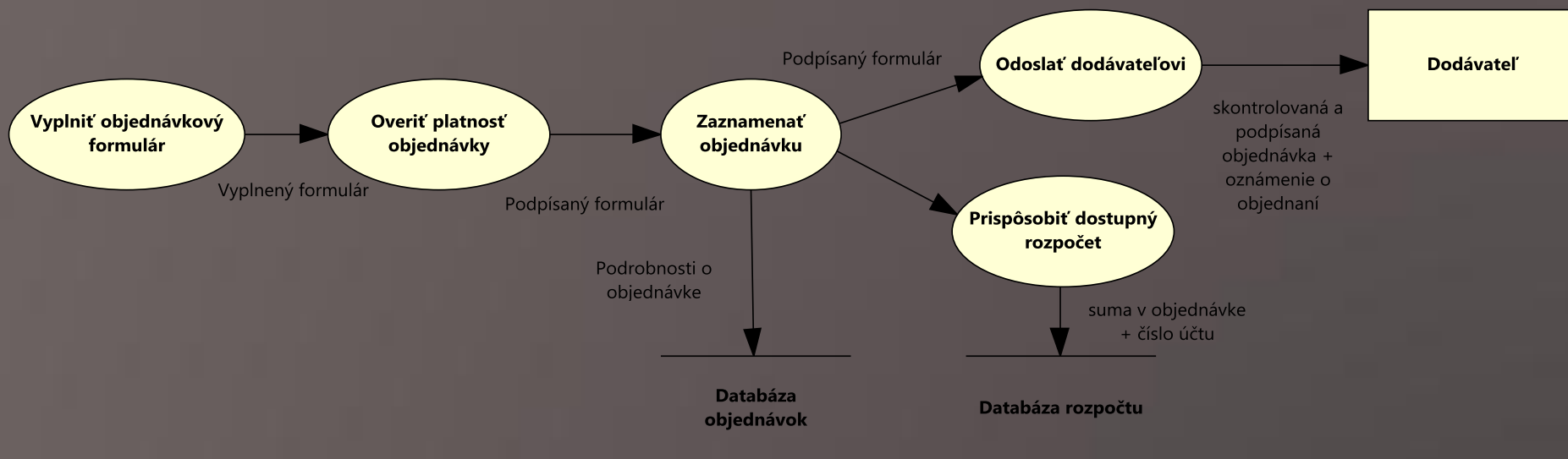
Behaviorálne modely

- použité na popis celkového správania systému
- ukážeme dva typy:
 - **modely spracovania dát** (data processing m.): ako sa dáta spracovávajú počas ich pohybu v systéme
 - **stavové modely** (state machine m.): ukazujú reakciu systému na udalosti
- oba modely sú nutnosťou pri popise správania sa systému

Behaviorálne modely: modely spracovania dát

- na modelovanie spracovania dát v systéme sa používajú diagramy toku dát
 - data flow diagrams
- ukazujú kroky, v ktorých sa dáta spracovávajú v systéme
- prirodzená súčasť mnohých analytických metód
- využíva sa jednoduchá notácia, ktorú dokáže pochopiť aj zákazník
- ukazuje celkový priebeh toku dát

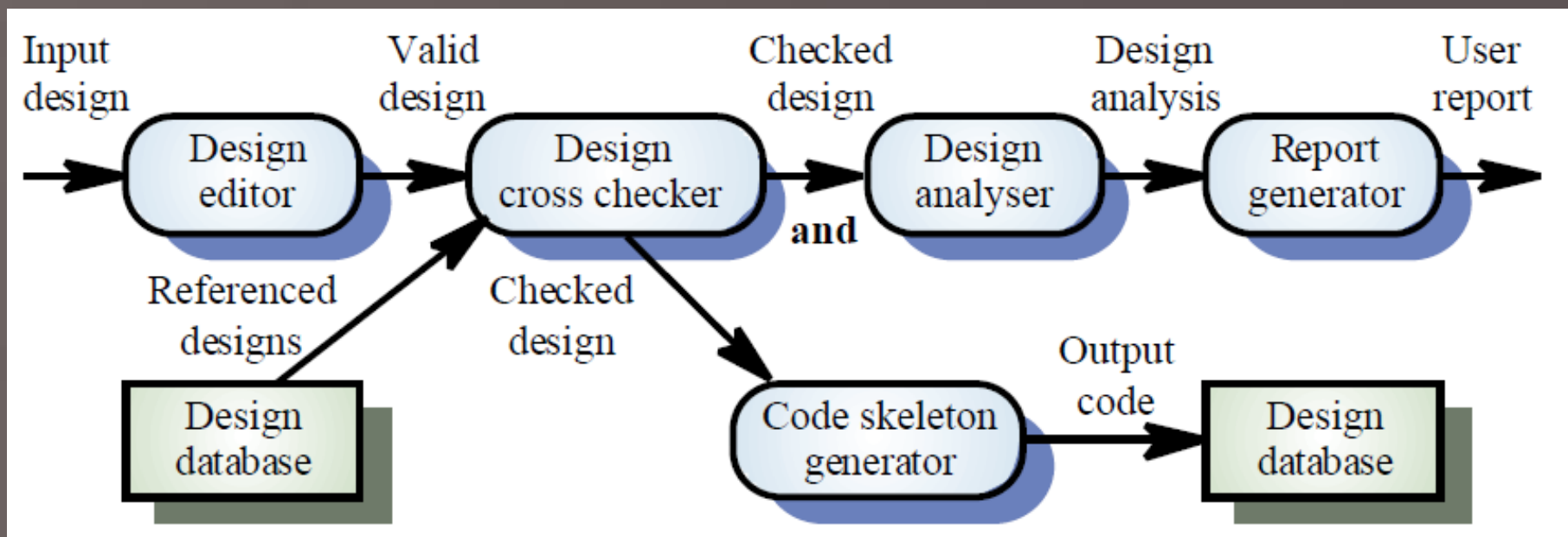
Príklad: diagram toku dát pre spracovanie objednávky



Behaviorálne modely: modely spracovávania dát: diagramy toku dát

- modelujú systém z hľadiska funkcionality
- sledovanie a dokumentovanie toku dát umožňuje lepšie pochopiť systém
- možno ich používať na znázornenie výmeny dát medzi systémom a ostatnými systémami

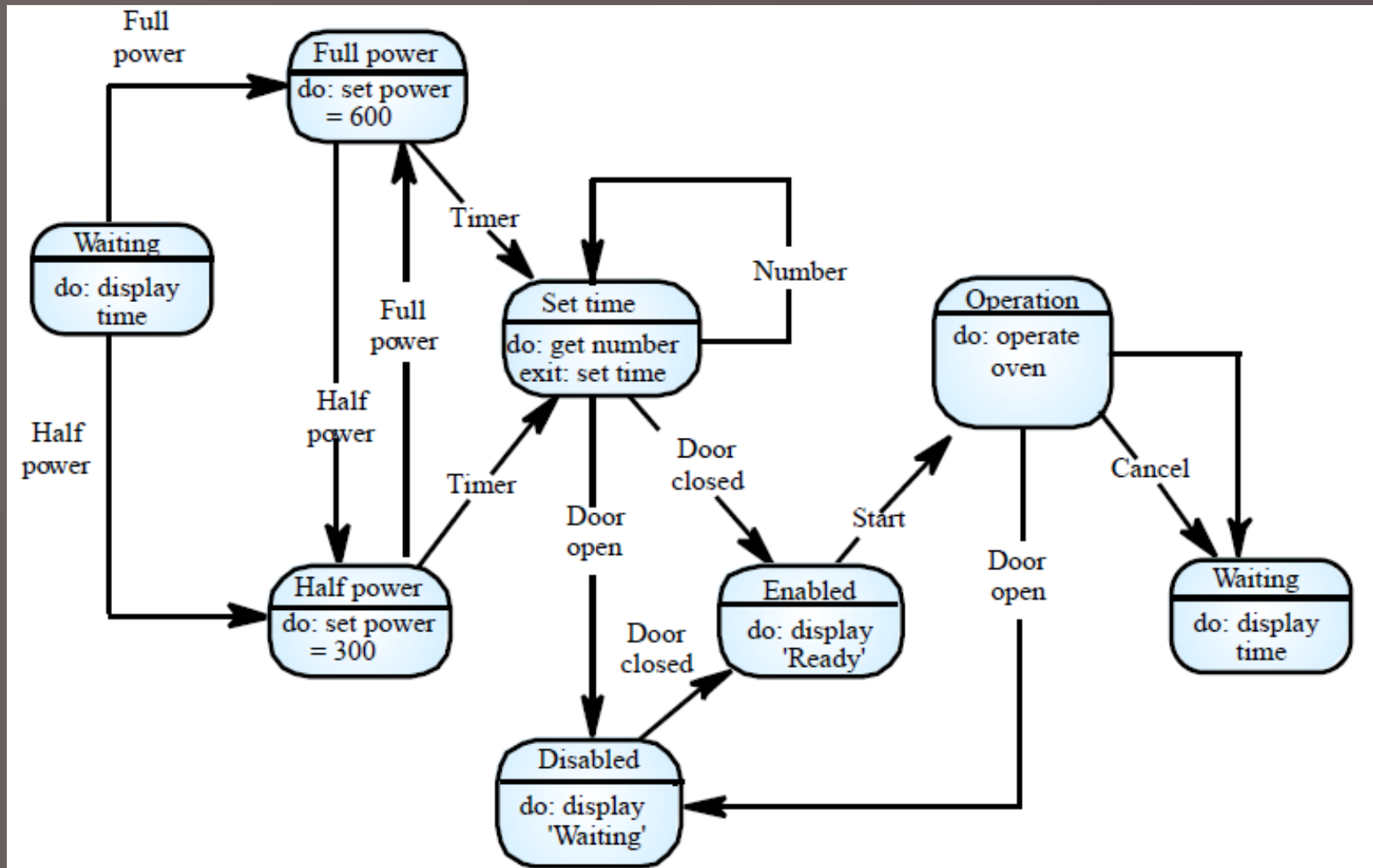
Príklad: diagram toku dát v CASE nástroji



Behaviorálne modely: modely spracovávania dát: stavové modely

- state machine models
- modelujú správanie systému vyplývajúce z reakcie na vonkajšie i vnútorné udalosti
- ukazujú reakciu systému na podnety
 - často používané pri modelovaní systémov fungujúcich v reálnom čase
- stav systému znázornený uzlami
- udalosti ako oblúky medzi stavmi
- UML: stavové diagramy (statecharts)

Stavový model mikrovlnky



Stavový model mikrovlnky

Stav	Popis
Čakajúca	Rúra čaká na vstup. Displej ukazuje aktuálny čas.
Polovičný výkon	Výkon nastavený na 300 W. Displej ukazuje "Polovičný výkon"
Plný výkon	Výkon nastavený na 600 W. Displej ukazuje "Plný výkon"
Nastavovanie času	Čas varenia sa určí z používateľovho vstupu. Displej ukazuje navolený čas varenia. Hodnota sa mení pri aktualizácii času.
Znefunkčnená	Kvôli bezpečnosti rúra prestala fungovať. Vnútorne svetlo svieti. Displej ukazuje "Nepripravená."
Pripravená	Rúra je pripravená na varenie. Vnútorne svetlo svieti. Displej ukazuje "Pripravená na varenie."
Pracujúca	Rúra pracuje. Svetlo svieti. Displej odpočíva čas. Po dokončení varenia sa ozve bzučiak trvajúci 5 sekúnd. Kým znie bzučiak, displej ukazuje "Varenie dokončené".

Podnety pre mikrovlnku

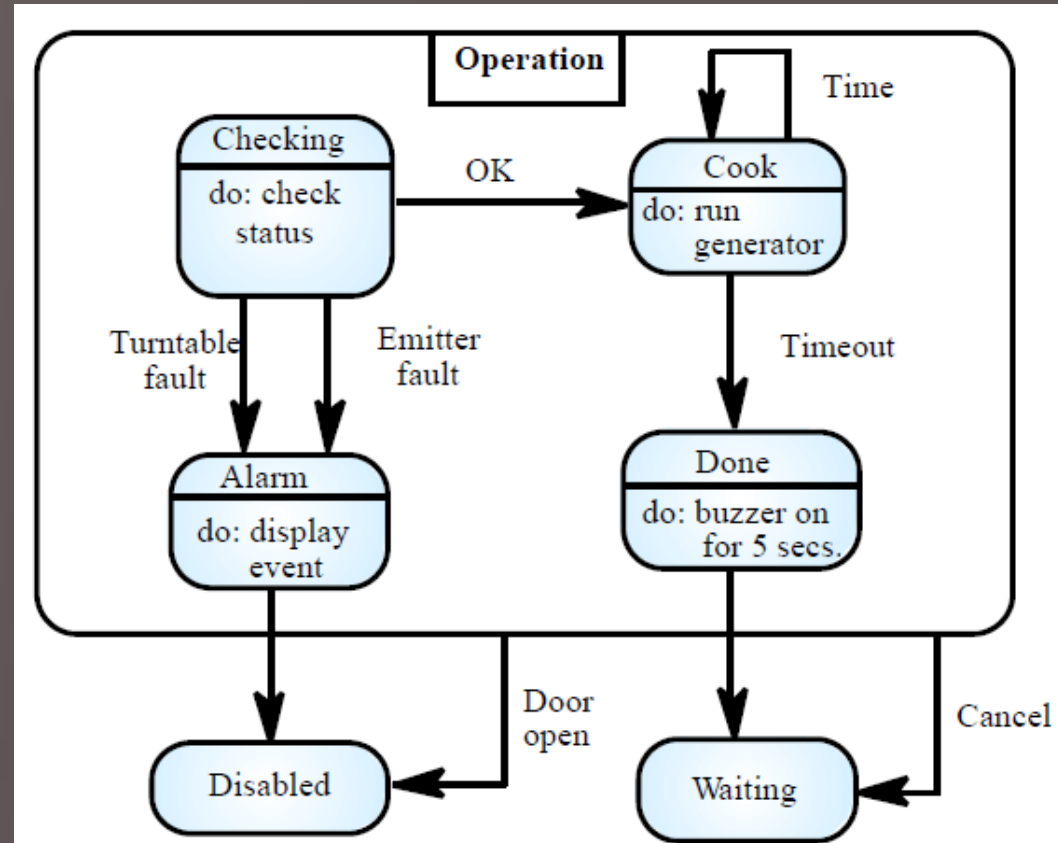
Podnet	Popis
Polovičný výkon	Používateľ stlačil tlačidlo "Polovičný výkon"
Plný výkon	Používateľ stlačil tlačidlo "Plný výkon"
Časovač	Používateľ stlačil niektoré z tlačidiel časovača
Číslo	Používateľ stlačil číslo.
Dvierka otvorené	Dvierka na rúre nie sú zatvorené.
Dvierka zatvorené	Dvierka na rúre sú zatvorené.
Štart	Používateľ stlačil tlačidlo Štart
Storno	Používateľ stlačil tlačidlo Storno.

Stavový diagram [statechart]

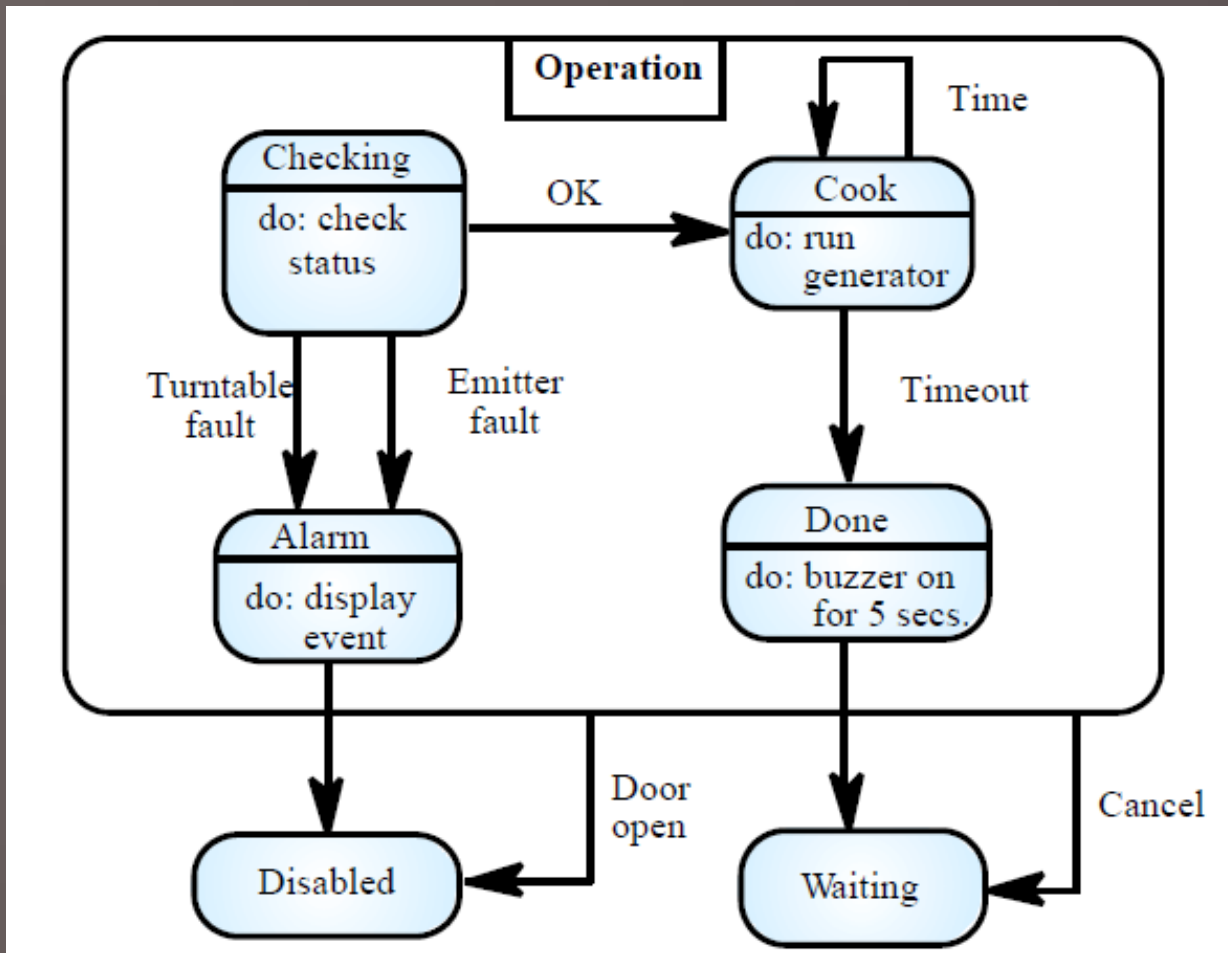
- za klauzulou **do** sa uvedie stručný popis udalosti, ktorá vedie k prechodu do iného stavu
- možno ho doplniť tabuľkami, ktoré popisujú stavy a podnety, ktoré vyvolali zmenu
- stav a podnet môžeme tiež bližšie popísať v dátovom slovníku (vid' ďalej)

Stavový diagram [statechart]

- počet stavov môže byť niekedy veľký
- môžeme zoskupiť niekoľko stavov do "superstavu" (nasledovný slajd: stav „Operácia“)
- superstav dekomponujeme v samostatnom modeli



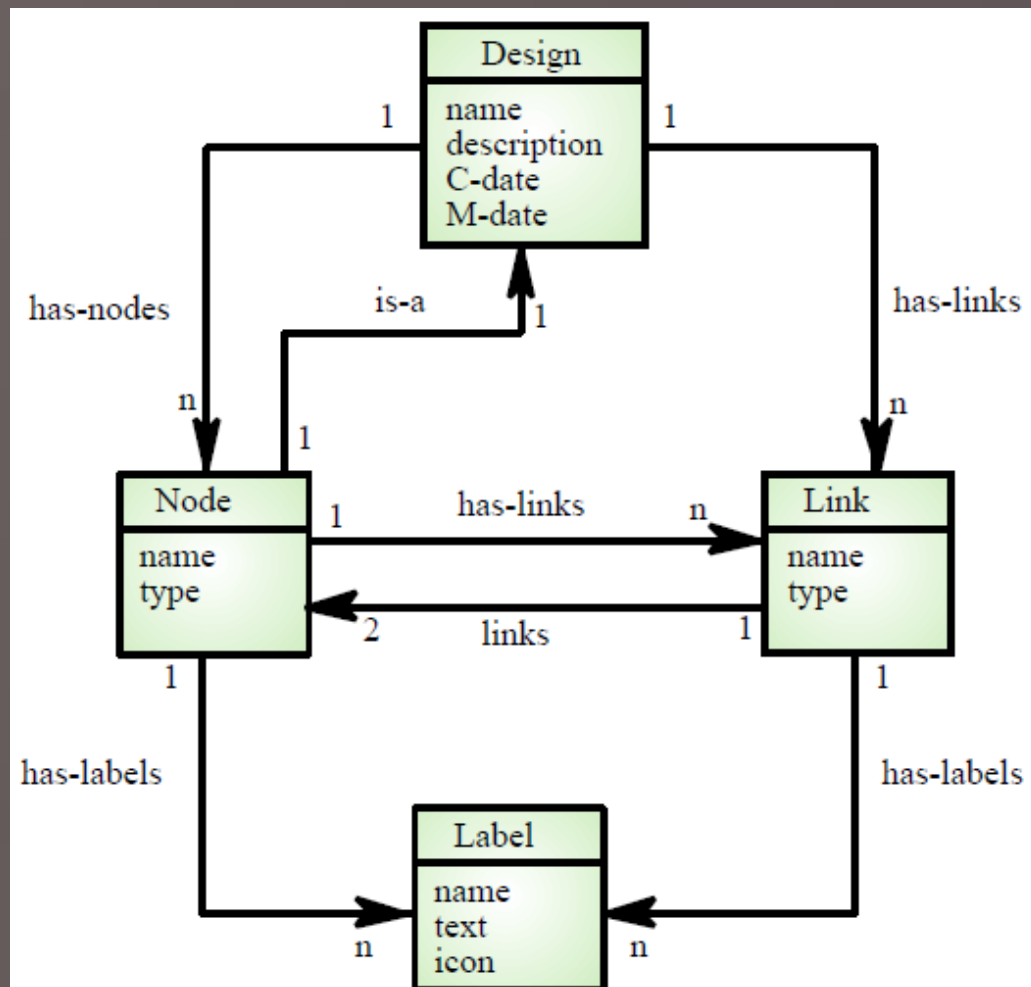
Príklad: stavový diagram mikrovlnky



Sémantické dátové modely

- používané na opis logickej štruktúry dát spracovávaných systémom
- **entitno-relačno-atribútový model** [entity-relation-attribute] udáva
 - entity systému
 - vzťahy medzi entitami
 - atribúty entít
- hojne používané v návrhu databáz
 - možno ich priamo implementovať v relačných databázach
- UML: nedefinuje žiadnu konkrétnu notáciu
 - možno použiť objekty a asociácie

Príklad: sémantický model pre návrh softvéru



Dátové slovníky [data dictionaries]

- zoznam všetkých termínov použitých v modeloch systému
- obsahuje tiež slovný popis entít, vzťahov a atribútov
- výhody:
 - evidencia terminológie
 - predchádza sa duplicitám
 - obsahuje organizačné znalosti, ktoré prepájajú analýzu, dizajn a implementáciu
- priamo podporované v CASE nástrojoch

Príklad: položky v dátovom slovníku

Názov	Popis	Typ	Dátum
Má-popisok	Relácia 1:N medzi entitami typu Uzol a Prepojenie a medzi entitami typu Popisok	vzťah	5. 10. 2008
Popisok	Obsahuje štruktúrovanú či neštruktúrovanú informáciu o uzloch a vzťahoch. Popisky sú znázornené ikonou (ktorá môže byť aj prázdna) a príslušným textom	entita	8. 10. 2008
Prepojenie	Relácia 1:1 medzi entitami dizajnu, ktoré sú znázornené ako uzly. Prepojenia sú typované a môžu byť pomenované.	vzťah	8. 12. 2008
Meno (popisku)	Každý popisok má svoje meno, z ktorého sa určí typ popisku. Meno musí byť jednoznačné vzhľadom na množinu typov popisov používaných v systéme	atribút	8. 12. 2008
Meno (uzla)	Každý uzol má meno, ktoré musí byť jednoznačné vzhľadom na návrh. Max. dĺžka: 64 znakov	atribút	15. 11. 2008

Objektové modely

- opisujú systém pomocou objektových tried
 - triedy známe z objektovo-orientovaného programovania
- objektová trieda je abstrakcia nad množinou objektov SO
 - spoločnými atribútmi
 - spoločnými službami (operáciami)
- možno vytvoriť viacero druhov objektových modelov
 - modely dedičnosti [inheritance m.]
 - agregáčné modely [aggregation m.]
 - modely interakcie [interaction m.]

Objektové modely

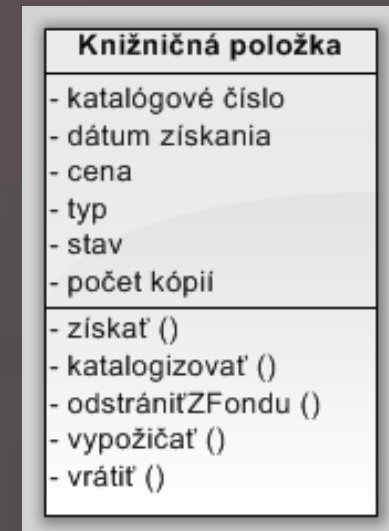
- prirodzený spôsob, ktorý odzrkadľuje entity zo skutočného sveta
 - systém manipuluje s týmito entitami
- potenciálne problémy
 - abstraktnejšie entity sa týmto spôsobom modelujú zložitejšie
 - identifikovanie objektových tried je považované za náročný proces, ktorý predpokladá hlboké znalosti o aplikačnej doméne
- výhoda: triedy, ktoré zodpovedajú entitám z aplikačnej domény možno použiť vo viacerých systémoch

Modely dedičnosti [inheritance m.]

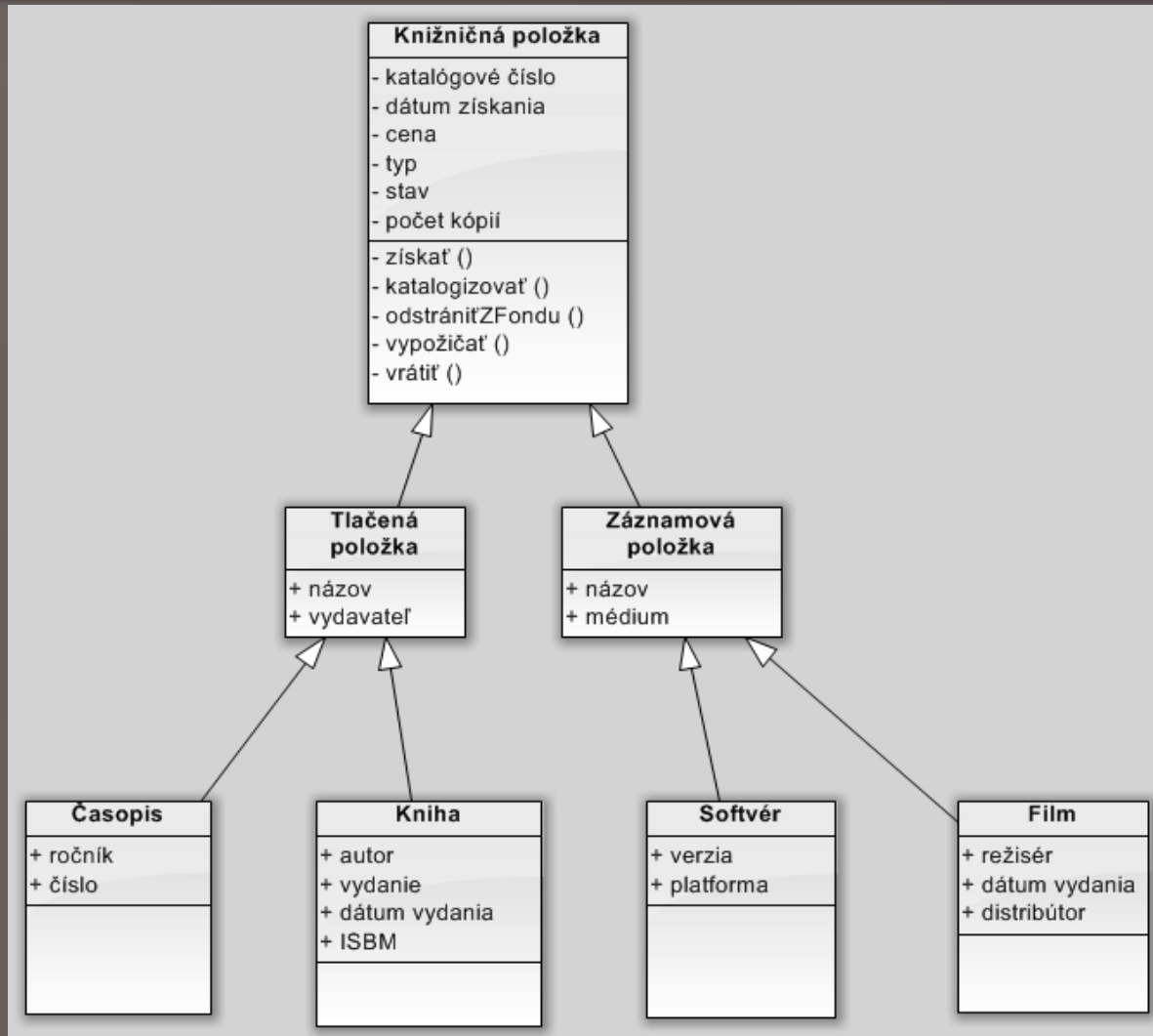
- objektové triedy sú v nich usporiadané do hierarchie
- triedy na vrchole hierarchie obsahujú spoločné rysy všetkých tried
- objektové triedy dedia atribúty a služby z jednej či viacerých nadtried
 - atribúty i triedy možno podľa potreby špecializovať
- návrh hierarchie môže byť náročný, ak chceme predísť duplicite v rozličných vetvách dedičnosti

Unified Modeling Language - UML

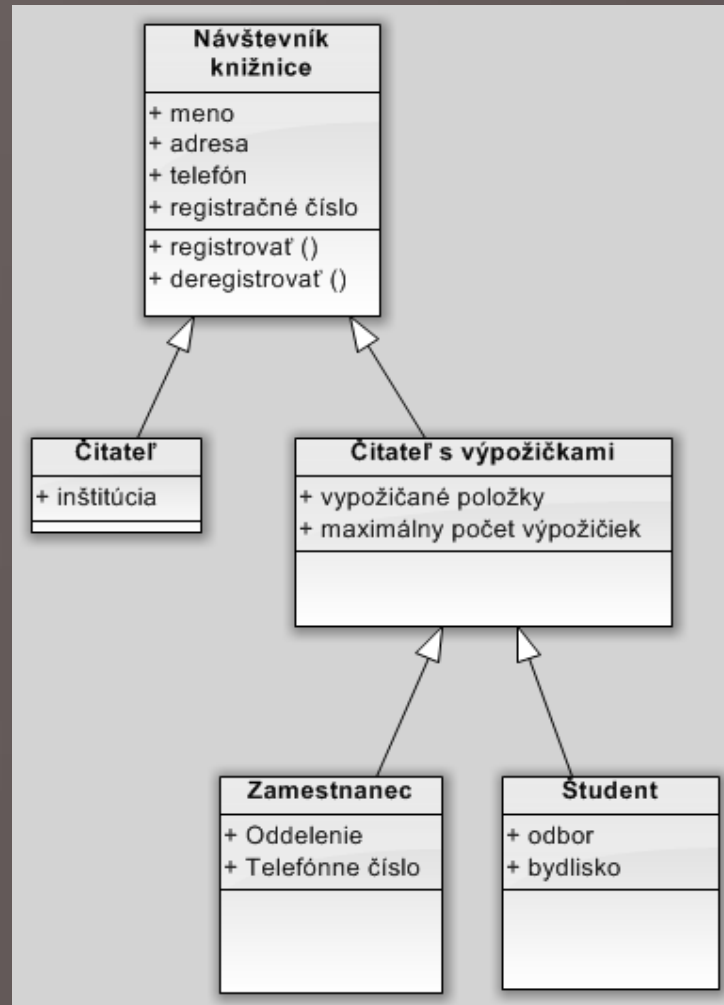
- modelovací jazyk
 - vytvorený vývojármi najčastejšie používaných metód objektovo-orientovanej analýzy a návrhu
- stal sa z neho štandard pre OO modelovanie
- notácia
 - objektové triedy sú obdĺžniky
 - na vrchu uvedený názov triedy
 - v strede atribúty
 - na spodku zoznam operácií



Hierarchia tried v knižnici



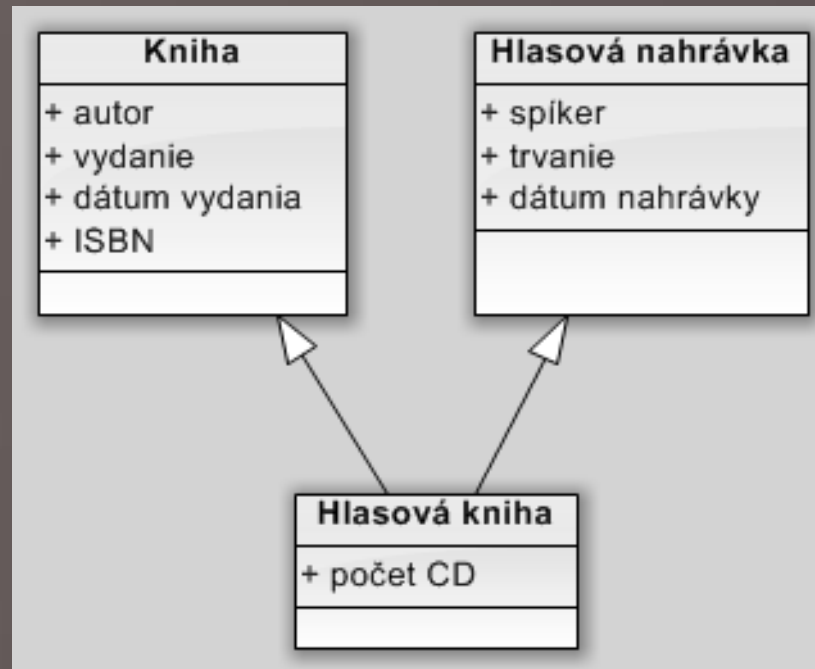
Hierarchia používateľov



Viacnásobná dedičnosť

- objektové triedy môžu v niektorých prípadoch dediť atribúty a služby z viacerých nadtried
- môže viesť k sémantickým konfliktom
 - atribúty/služby s rovnakým menom z rôznych nadtried môžu mať rozličnú sémantiku
 - smrtníci smaragd smrti (deadly diamond of death)
- reorganizácia hierarchia tried býva zložitejšia
- moderné programovacie jazyky zámerne zakazujú viacnásobnú dedičnosť

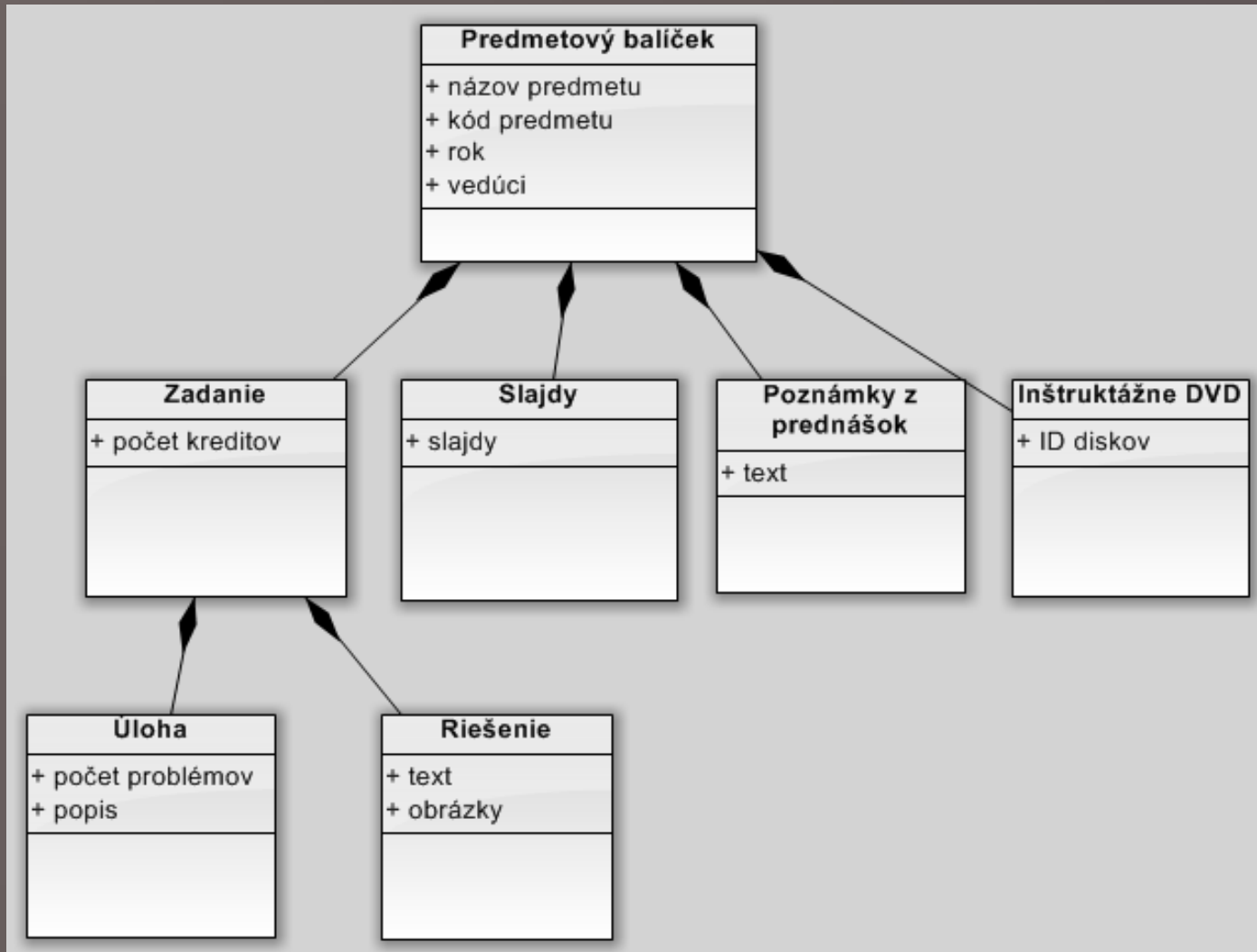
Príklad viacnásobnej dedičnosti



Agregácia objektov

- znázorňuje spôsob, ktorým sú triedy zodpovedajúce skupinám objektov, zložené z iných tried
- analogické ku vzťahu "part-of" ("je súčasťou") v sémantických dátových modeloch

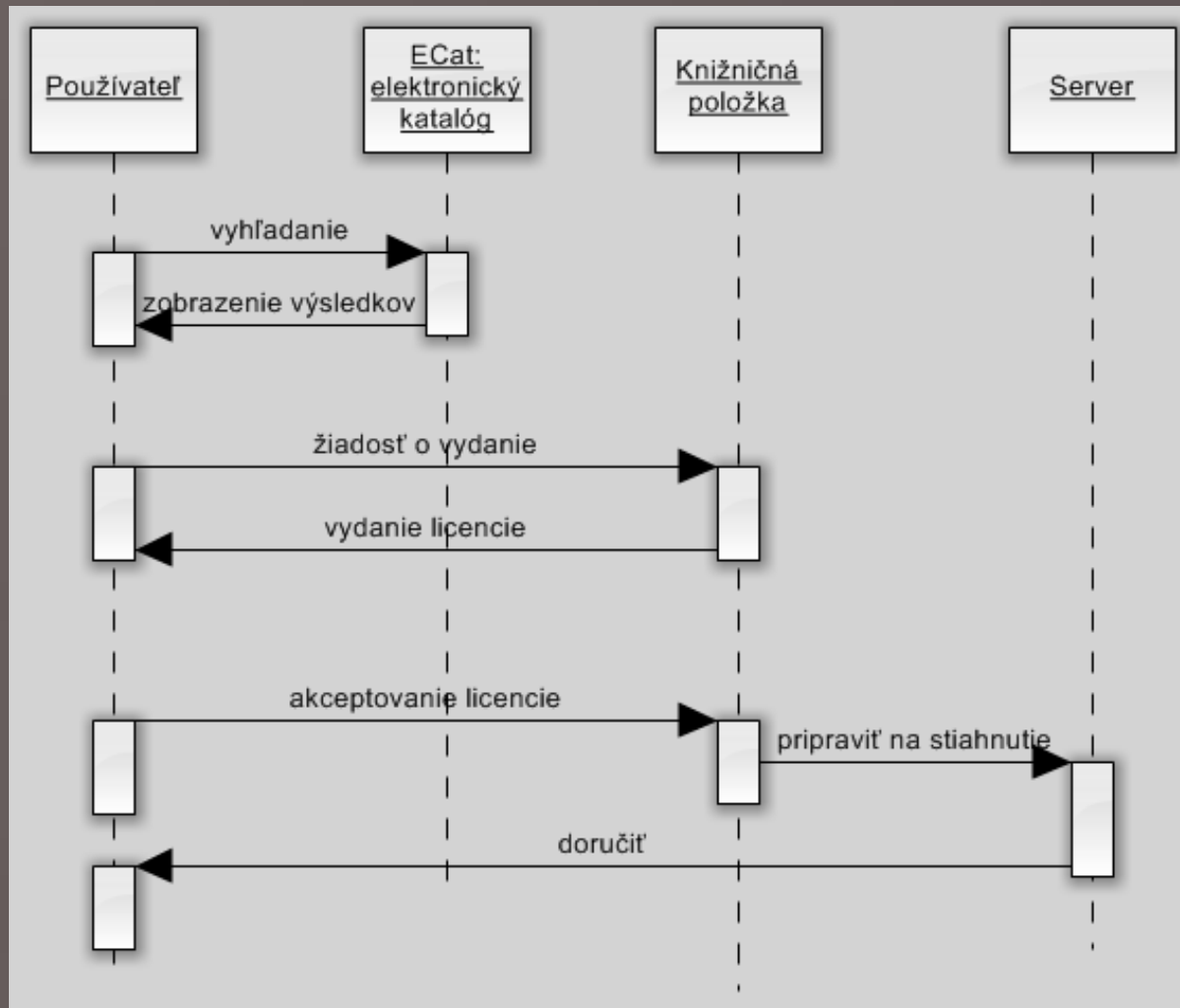
Príklad: agregácia objektov



Modelovanie správania objektov

- behaviorálny model [behavioural m.] znázorňuje interakcie medzi objektami
- interakciami sa dosiahne konkrétne správanie systému definované v rámci use-case
- UML:
 - sekvenčné diagramy (sequence diagrams)
 - diagramy spolupráce (collaboration diagrams)

Príklad: vydanie elektronického materiálu



CASE nástroje

- CASE – computer aided software engineering
 - počítačom podporované softvérové inžinierstvo
- sada nástrojov navrhnutých pre podporu aktivít súvisiacich so softvérovým procesom
 - analýza, dizajn, testovanie
- nástroje pre analýzu a dizajn podporujú systémové modelovanie
 - počas zbierania požiadaviek (requirements engineering)
 - počas návrhu systému
- môžu byť zamerané na konkrétnu návrhovú metódu
- ale môžu podporovať tvorbu rozličných typov modelu systému

Nástroj pre analýzu a dizajn



Kľúčové body

- **model** je abstraktný pohľad na systém
 - jednotlivé typy modelov sa vzájomne dopĺňajú
 - každý z nich demonštruje jeden pohľad na systém
- **kontextové modely** ukazujú postavenie systému v prostredí vzhľadom k iným systémom a procesom
- **modely toku dát** [data flow m.] možno použiť na modelovanie spracovania dát v systéme
- **stavové modely** [state machine m.] ukazujú správanie systému, ktoré vyplýva z vnútorných či vonkajších udalostí

Klíčové body

- **sémantické datové modely** ukazujú logickú štruktúru dát, ktoré systém importuje a exportuje
- **objektové modely** opisujú logické entity v systéme, ich klasifikáciu a agregáciu
- CASE nástroje podporujú vývoj modelov systému

ĎAKUJEM ZA POZORNOST